

CLAIMS

Having thus described our invention, what we claim as new and desire by Letters Patent is as follows:

- 1 1. A method of accessing data in a non-relational database, the method
2 comprising the steps of:
3 creating a master view having a master view index referencing the data;
4 creating a subordinate view of the master view having a subordinate view index
5 referencing a subset of said master view index, where the subordinate view defines
6 accessible portions of the data and the subordinate view index is linked to a subset of the
7 master view index; and
8 accessing the data via the subordinate view.
- 1 2. The method of claim 1, wherein the creating a master view includes defining at least
2 one of sorted and categorized columns associated with the master view.
- 1 3. The method of claim 1, wherein the creating a subordinate view step includes
2 defining at least one of a collapsed subordinate view and a non-collapsed subordinate
3 view.
- 1 4. The method of claim 1, further comprising automatically managing the subordinate
2 view.
- 1 5. The method of claim 1, wherein the accessing step includes creating an index map
2 which links the accessible data associated with the subordinate view to the master index.

- 1 6. The method of claim 5, further comprising caching at least one of the subordinate
2 view and the temporary index map, wherein the caching step includes:
3 checking whether a predetermined time period has elapsed by checking an
4 elapsed time period counter;
5 if elapsed, checking whether access frequency exceeds a predetermined
6 threshold by checking an access counter for the subordinate view; and
7 if the predetermined threshold is exceeded, checking whether the at least one of
8 the subordinate view and the index map can be cached,
9 if so, then caching at least one of the subordinate view and the temporary index.
- 1 7. The method of claim 6, further comprising resetting one of the elapsed time period
2 counter to start a new elapsed time period for counting access frequencies and the access
3 counter for counting access frequencies to the temporary index map during the new
4 elapsed time period.
- 1 8. The method of claim 1, further comprising maintaining historical information
2 including access frequency to the subordinate view.
- 1 9. The method of claim 1, wherein the accessing step provides one of a reduction of
2 data transferred to a client in a client-server architecture, a decrease in the amount of data
3 manipulated during the accessing step, a decrease in response time to an access request,
4 an increased performance, and a decrease in index size.
- 1 10. The method of claim 1, wherein the data includes at least one of categorized non-
2 hierarchical data., hierarchical data, and categorized hierarchical data.
- 1 11. The method of claim 1, wherein the master view has a master index referencing at
2 least a portion of the hierarchical data in the non-relational database.

- 1 12. The method of claim 1, wherein the creating a subordinate view step includes creating
2 a plurality of subordinate views associated with one or more master views.
- 1 13. A method of enhancing performance when accessing hierarchical data in a non-
2 relational database, the method comprising the steps of:
3 creating at least one subordinate view having a subordinate index referencing a
4 subset of a master index of at least one master view;
5 creating a subordinate view index map associated with the at least one subordinate
6 view when accessing the hierarchical data; and
7 accessing the at least a portion of hierarchical data by using the subordinate view
8 index map,
9 wherein an amount of data accessed using the at least one subordinate view is less
10 than the amount of data when accessing the at least one master view.
- 1 14. The method of claim 13, wherein the creating at least one subordinate view includes
2 defining at least one of sorted and categorized columns associated with the at least one
3 master view.
- 1 15. The method of claim 13, further comprising maintaining historical information
2 including access frequency to the subordinate view.
- 1 16. The method of claim 15, further comprising caching at least one of the at least one
2 subordinate view, the subordinate view index map and temporary index, wherein the
3 caching step includes:
4 checking whether a predetermined time period has elapsed by checking an
5 elapsed time period counter; and
6 if elapsed, checking whether the access frequency exceeds a predetermined
7 threshold by checking a access counter for the at least one subordinate view;

8 if the predetermined threshold is exceeded, checking whether the at least one
9 subordinate view can be cached,
10 if so, caching at least one of the at least one subordinate view, the subordinate
11 view index map, and the temporary index.

1 17. The method of claim 13, further including assigning priorities to one of the at least
2 one master view and at least one subordinate view to grade performance.

1 18. The method of claim 13, wherein the subordinate view includes at least one of a
2 collapsed view and a non-collapsed view and access via the collapsed view providing less
3 data than access via a non-collapsed view.

1 19. An apparatus for accessing hierarchical data in a non-relational database, the
2 apparatus comprising:
3 a first component to create a master view having a master view index referencing
4 hierarchical data;
5 a second component to create a subordinate view of the master view that has a
6 subordinate view index that references a subset of said master view index, the
7 subordinate view defines accessible data of the hierarchical data and the subordinate view
8 index linked to a subset of the master view index; and
9 a third component to access the hierarchical data via the subordinate view.

1 20. A computer program product comprising a computer usable medium having readable
2 program code embodied in the medium, the computer program product includes:
3 a first computer program code to create a master view having a master index
4 referencing the hierarchical data;
5 a second computer program code to create a subordinate view having a
6 subordinate index referencing a subset of said master index, where the subordinate view

7 defines accessible data of the hierarchical data and the subordinate view linked to a
8 subset of the master view index;
9 a third computer program code to access the hierarchical data via the subordinate
10 view; and
11 a fourth computer program code to create an index map and temporary index for
12 linking data associated with the subordinate view to the master index.